Lab Assignment
& Solution

Cybersecurity Professional Program

Introduction to Python
for Security

File System & Error Handling

PY-04-LS7
Encoding & Decoding the Secret

Note: Solutions for the instructor are shown inside the green box.

Lab Objective

Become familiar with encoding libraries and practice encoding strings.

Lab Mission

Encode and decode a Base64 secret using Python and print the original message to the console.

Lab Duration

15-20 minutes

Requirements

- Knowledge of how to handle input from users, and variables.
- Working knowledge of Base64 modules.
- Basic knowledge of the Unicode and Byte data types.

Resources

- Environment & Tools
    - Windows, MacOS, Linux
        - Python 3
        - PyCharm
- Extra Lab Links:
    - https://docs.python.org/3/library/base64.html
    - https://docs.python.org/3/howto/unicode.html

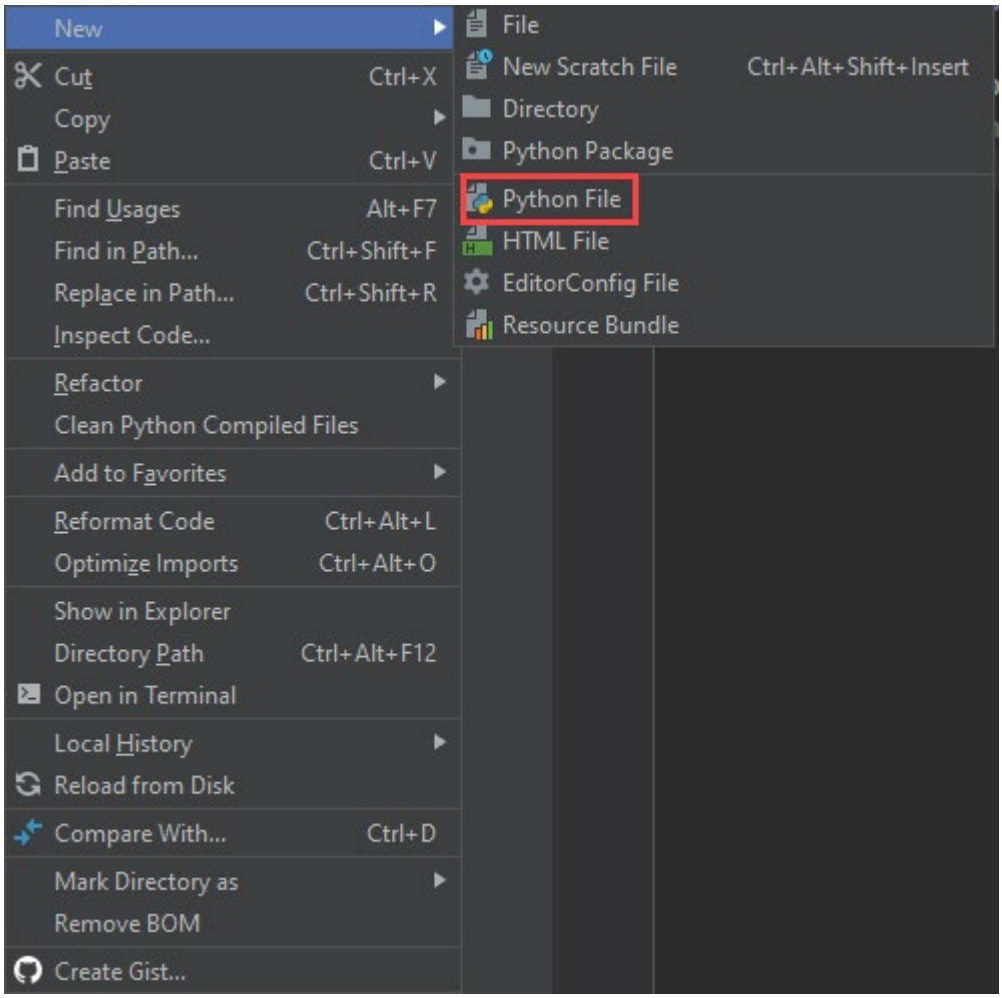
Textbook References

- Chapter 4: File System and Error Handling

Lab Task

Write a Python script that asks the user to encode a message in Base64. The script should then decode the message and print the results on the console.

1. Create a new Python file in PyCharm by right-clicking the project you created and selecting **New → Python File**.



2. Import the Base64 module.

```
import base64
```

3 Ask the user to provide a secret, and assign it to a variable.

```
import base64

secret = input("Please enter a secret: ")
```

4 Encode the secret using b64encode() and then convert the Byte data type returned by b64encode() to UTF-8 encoding. Save the output to a new variable.
Note: The aim is to encode the base64 secret, but the secret will also need to be encoded to UTF-8 format to obtain a readable message required by the base64 module.

```
import base64

secret = input("Please enter a secret: ")
encoded = base64.b64encode(secret.encode("utf-8"))
```

5 Decode the secret using b64decode(), and save the output to a new variable.

```
import base64

secret = input("Please enter a secret: ")
encoded = base64.b64encode(secret.encode("utf-8"))
decoded = base64.b64decode(encoded)
```

6 Print the encoded secret.
Note that the output is “gibberish” (unreadable), which is what we want.

```
import base64

secret = input("Please enter a secret: ")
encoded = base64.b64encode(secret.encode("utf-8"))
decoded = base64.b64decode(encoded)
print("The encoded string is ", encoded)
```

7 Print the decoded secret.
Note: The output will show a b'{input_string}'. Suppose the string you entered was 'p@wn3d' in Step 2. You would then see an output of b'p@wn3d'.
The string inside the single quotes after the "b" is the decoded string.

```
import base64

secret = input("Please enter a secret: ")
encoded = base64.b64encode(secret.encode("utf-8"))
decoded = base64.b64decode(encoded)
print("The encoded string is ", encoded)
print("The decoded string is ", decoded)
```